APPLICATION

FOR

UNITED STATES LETTERS PATENT

TITLE: AUTOMATIC ISOLATION IN LOOPS

APPLICANT: KARL M. HENSON, DAVID BREWER, HOSSEIN HASHEMI AND DAVID BALDWIN
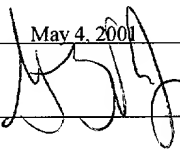
AUTOMATIC ISOLATION IN LOOPS

This is a divisional of U.S. application serial no. 09/398,523, filed September 15, 1999 (allowed), which is a divisional of U.S. application serial no. 09/071,678, filed May 1, 1998 (issued).

TECHNICAL FIELD

5   The present invention relates to electronic network communications systems, and more specifically to automatic isolation of a node or loop segment in a loop network where a data channel transmitting data from a hub port to the node or loop segment has failed.

BACKGROUND OF THE INVENTION

10   Electronic data systems are frequently interconnected using network communication systems. Area-wide networks and channels are two approaches that have been developed for computer network architectures. Traditional networks (*e.g.*, LAN's and WAN's) offer a great deal of flexibility and relatively large distance capabilities. Channels, such as the Enterprise System Connection (ESCON) and the Small Computer System Interface (SCSI), have been

15   developed for high performance and reliability. Channels typically use dedicated short-distance connections between computers or between computers and peripherals.

Features of both channels and networks have been incorporated into a new network standard known as "Fibre Channel". Fibre Channel systems combine the speed and reliability of channels with the flexibility and connectivity of networks. Fibre Channel products

20   currently can run at very high data rates, such as 266 Mbps or 1062 Mbps. These speeds are sufficient to handle quite demanding applications, such as uncompressed, full motion, high-quality video. ANSI specifications, such as X3.230-1994, define the Fibre Channel network. This specification distributes Fibre Channel functions among five layers. The five functional layers of the Fibre Channel are: FC-0 - the physical media layer; FC-1 - the coding

25   and encoding layer; FC-2 - the actual transport mechanism, including the framing protocol and flow control between nodes; FC-3 - the common services layer; and FC-4 - the upper layer protocol.

There are generally three ways to deploy a Fibre Channel network: simple point-to-point connections; arbitrated loops; and switched fabrics. The simplest topology is the point-to-point configuration, which simply connects any two Fibre Channel systems directly. Arbitrated loops are Fibre Channel ring connections that provide shared access to bandwidth via arbitration. Switched Fibre Channel networks, called "fabrics", are a form of cross-point switching.

Conventional Fibre Channel Arbitrated Loop ("FC-AL") protocols provide for loop functionality in the interconnection of devices or loop segments through node ports. However, direct interconnection of node ports is problematic in that a failure at one node port in a loop typically causes the failure of the entire loop. This difficulty is overcome in conventional Fibre Channel technology through the use of hubs. Hubs include a number of hub ports interconnected in a loop topology. Node ports are connected to hub ports, forming a star topology with the hub at the center. Hub ports which are not connected to node ports or which are connected to failed node ports are bypassed. In this way, the loop is maintained despite removal or failure of node ports.

More particularly, an FC-AL network is typically composed of two or more node ports linked together in a loop configuration forming a single data path. Such a configuration is shown in FIG. 1A. In FIG. 1A, six node ports 102, 104, 106, 108, 110, 112 are linked together by data channels 114, 116, 118, 120, 122, 124. In this way, a loop is created with a datapath from node port 102 to node port 104 through data channel 114 then from node port 104 to node port 106 through data channel 116, and so on to node port 102 through data channel 124.

When there is a failure at any point in the loop, the loop datapath is broken and all communication on the loop halts. FIG. 1B shows an example of a failure in the loop illustrated in FIG. 1A. Data channel 116 connecting node port 104 to node port 106 has a failure 125 before entering node port 106. The failure 125 could be caused by a problem such as a physical break in the wire or electromagnetic interference causing significant data corruption or loss at that point. Node port 106 no longer receives data or valid data from node port 104 across data channel 116. At this point, loop 100 has been broken. Data no longer flows in a circular path and the node ports are no longer connected to one another. For

example, node port 104 cannot transmit data to node port 108 because data from node port 104 does not pass node port 106. The loop has, in effect, become a unidirectional linked list of node ports.

In a conventional FC-AL system, recovery proceeds according to a standard. Accordingly, when node port 106 detects that it is no longer receiving valid data across data channel 116, node port 106 begins to generate loop initialization primitive ("LIP") ordered sets, typically LIP (F8, AL_PS) or LIP (F8, F7) ("LIP F8") ordered sets. "AL_PS" is the arbitrated loop address of the node port which is issuing the LIP F8 ordered sets, in this case, node port 106. The LIP F8 ordered sets propagate around the loop. Each node receiving a LIP F8 primitive sequence stops generating data or other signals and sends a minimum of 12 LIP F8 ordered sets. A sequence of three consecutive LIP F8 ordered sets forms a LIP F8 primitive sequence. At this point, the LIP F8 primitive sequences and ordered sets composing primitive sequences propagate through the broken loop 100 shown in FIG. 1B. Loop 100 typically does not function again until the data channel 116 has been repaired or replaced, such as by physical replacement or bypass by a second wire or cable. When node port 106 receives the LIP F8 primitive sequence, node port 106 begins loop initialization.

A conventional partial solution to recovery from a broken node port-to-node port loop is provided by the introduction of a hub within a loop. A hub creates a physical configuration of node ports in a star pattern, but the virtual operation of the node ports continues in a loop pattern. The connection process (*i.e.*, sending data between node ports) and interaction with the hubs is effectively transparent to the node ports connected to the hub which perceive the relationship as a standard FC-AL configuration.

FIG. 2A illustrates an arbitrated loop 200 with a centrally connected hub. Similar to loop 100 illustrated in FIG. 1A and 1B, loop 200 includes six node ports 202, 204, 206, 208, 210, 212, each attached to a hub 214. Hub 214 includes six hub ports 216, 218, 220, 222, 224, 226 where each hub port is connected to another hub port in a loop topology by a sequence of internal hub links. In this way, node ports 202 - 212 are each connected to a corresponding hub port 216 - 226. Thus, node ports 202 - 212 operate as though connected in a loop fashion as illustrated in FIG. 1A.

When a failure occurs on a data channel carrying data from a node port to a hub port, the loop is maintained by bypassing the failed node port. In a conventional hub, when a hub port no longer receives data from a node port, the hub port goes into a bypass mode where, rather than passing data received on the data channel from the node port, the hub port passes data received along the internal hub link from the previous hub port. For example, data channel 234 connecting node port 206 to hub port 220 may fail, such as through physical disconnection or interference such that valid data no longer passes from node port 206 to hub port 220. Hub port 220 detects the cessation of valid data from node port 206 and enters bypass mode. In this way, the loop integrity is maintained. Rather than breaking the loop, as was the case illustrated in FIG. 1B, the bypass mode of a hub port allows the loop to be preserved. As shown in FIG. 2A, data continues to flow around the loop even while data channel 234 has failed because hub port 220 is operating in a bypass mode and isolates node port 206.

FIG. 2B illustrates a different problem which is unresolved by conventional hub technology. In FIG. 2B, a data channel 236 carrying data from hub port 220 to node port 206 has failed. In this case, hub port 220 continues to receive data from node port 206 along data channel 234. Because node port 206 is no longer receiving data from the loop, node port 206 under conventional FC-AL protocols typically detects the link failure and begins to generate LIP F8 ordered sets. The hub ports of a conventional hub 214 cannot differentiate the type of signal being received from an attached node port. As a result, in the situation illustrated in FIG. 2B, hub port 220 does not recognize the LIP F8 sequence being received from node port 206 as anything different from the standard data received from node port 206. Thus, hub port 220 does not enter a bypass mode, and sends the data from node port 206 to hub port 222. As the LIP F8 ordered sets continue to be sent by node port 206, they form a LIP F8 primitive sequence, as described above. When the other node ports in the loop receive the LIP F8 primitive sequence, those nodes cease ordinary data processing and transmission and begin to generate LIP F8 ordered sets. At this point, while the virtual nature of the loop could be maintained through a bypass of the failed node port, because a conventional hub port such as hub port 220 does not recognize the LIP F8 nature of the data being sent from the connected

node port 206, a situation similar to that illustrated in FIG. 1B results. LIP F8 ordered sets propagate around the loop until all node ports are attempting loop initialization.

The inventors have determined that it would be desirable to provide a hub port that can create an automatic bypass upon detection of a LIP F8 primitive sequence from an attached

5   node port.

SUMMARY

The preferred embodiment of the invention provides a hub port in a hub of a loop network which automatically bypasses a node port which is generating a particular loop failure initialization sequence. The hub port contains a detection circuit which enables the

10   hub port to detect loop failure initialization data received from its attached node port. Upon detecting such data from an attached node port, the hub port replaces such data with buffer data to be passed to the next hub port. Upon detecting the completion of a loop failure initialization sequence from an attached node port, the hub port enters a bypass mode. The hub port no longer passes on output from its attached node port and instead forwards along

15   the internal hub link data received from the previous hub port in the hub loop. The bypass is maintained until the hub port receives a primitive sequence indicating the recovery of the attached node port.

One embodiment provides a hub port in a hub of a Fibre Channel arbitrated loop which automatically bypasses a node port which is generating a LIP F8 primitive sequence. The

20   hub port of the preferred embodiment contains a LIP detection circuit which enables the hub port to detect the generation of LIP F8 ordered sets by its attached node port. Upon receiving a LIP F8 ordered set from an attached node port, a hub port of a preferred embodiment generates fill words to be passed to the next hub port. Upon the completion of a LIP F8 primitive sequence from an attached node port, the hub port of the preferred embodiment

25   enters a bypass mode and no longer passes on output from its attached node port and instead forwards data received along the internal hub link from the previous hub port in the hub loop. The bypass is maintained until the hub port receives a LIP primitive sequence other than a LIP F8 primitive sequence from the attached node port.

-5-

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1A shows a prior art loop of directly interconnected node ports.

FIG. 1B shows a prior art loop including a failed data channel.

FIG. 2A shows a prior art loop including a hub.

FIG. 2B shows a prior art loop including a hub where a data channel has failed.

FIG. 3 shows a block diagram of a hub port of the preferred embodiment.

DETAILED DESCRIPTION

The preferred embodiment provides a mechanism to automatically bypass a node port or loop segment attached to a hub port, where the node port or loop segment is sending loop failure initialization sequences, such as LIP (F8, AL_PS) or LIP (F8, F7) ("LIP F8") primitive sequences. The invention is explained below in the context of a Fibre Channel Arbitrated Loop ("FC-AL") network as an illustration of the preferred embodiment. However, the invention may have applicability to networks with similar characteristics as FC-AL networks.

If a data channel carrying data to a node port or loop segment from a network hub port develops a link failure, the node port or loop segment is isolated from the hub loop and the other node ports on the hub loop are able to continue communication while the failed node port or loop segment is isolated from the loop.

The preferred embodiment provides a hub port which detects failures in its connection to a node port by detecting loop failure initialization sequences generated by the node port. The hub port then isolates the node port, allowing the remainder of the loop to function with the link error removed, hidden by the bypass mode of the hub port.

When a hub port of the preferred embodiment receives loop failure initialization data from the attached node port, the hub port does not pass the loop failure initialization data along the loop to the next hub port. The hub port replaces the loop failure initialization data with buffer data which is sent to the next hub port in the loop. If a loop failure initialization sequence is received (*i.e.*, some specified combination of loop failure initialization data), then the source of the loop failure initialization data (*i.e.*, the node port or loop segment which is generating the loop failure initialization data) is isolated until a loop initialization primitive

sequence other than a loop failure initialization sequence is received from the node port and detected at the hub port.

For example, in an FC-AL implementation, when a hub port receives LIP F8 ordered sets from the attached node port, the hub port replaces the LIP F8 ordered set with a "current fill word". If a LIP F8 primitive sequence (*e.g.*, three consecutive identical LIP F8 ordered sets), is effected, then the node port or loop segment which is generating the LIP F8 ordered sets is bypassed. The bypass is maintained until the hub port receives a LIP primitive sequence other than a LIP F8 primitive sequence.

Fill words are used under conventional FC-AL protocols as buffers between data frames. Data received from a node port is typically temporarily stored in a buffer within the hub port. The data typically leaves the buffer in a first in, first out manner ("FIFO"). The data rate of output from the hub port is not necessarily the same as the data rate of input from the node port. As a result, the data in the buffer may "run dry" if the data rate of the node port is slower than the data rate of the hub port. Conventional FC-AL protocols solve this problem by supplying inter-frame fill words when the data in the buffer supplied from the node port is low. Thus, fill words are used to maintain continuity of the data stream along the loop. Typically a sequence of six fill words is used between frames. However, hub ports and node ports may add or delete from the number of fill words present to maintain data integrity as determined by the FC-AL protocols. A continuous stream of data alone is improper under FC-AL protocols and may be a protocol violation error. The "current fill word" is a fill word defined by FC-AL protocols, and may vary depending upon loop traffic. Accordingly, the generation of fill words by the hub port which is receiving LIP F8 ordered sets from the attached node port is consistent with conventional FC-AL protocols.

Under current FC-AL protocols, a LIP F8 primitive sequence includes three consecutive identical LIP F8 ordered sets. Pursuant to the invention in an FC-AL implementation, the bypass of a node port does not occur until a LIP F8 primitive sequence has been received by the hub port. Upon receiving a first LIP F8 ordered set from an attached node port, the hub port "consumes" that LIP F8 ordered set and instead passes a current fill word to the next hub port. If the hub port receives a second consecutive identical LIP F8 ordered set, the hub port

again substitutes the current fill word for transmission to the next hub port. If not, the hub port passes along that properly formatted data and returns to normal operation.

If a third consecutive identical LIP F8 ordered set is received by the hub port from the attached node port, the hub port recognizes that a LIP F8 primitive sequence has been received and that the associated node port has failed. At this point, the hub port enters a bypass mode and passes along data from the previous hub port in the loop to the next hub port. This bypass is a similar operation to when the hub port is not attached to a node port at all. In that case, the hub port is also in a bypass mode (for example, where a hub containing n hub ports is connected to some number of node ports less than n). Those hub ports which are not attached to node ports are in a bypass mode and relay information from the previous hub port to the next hub port.

A hub port which has entered bypass mode due to the reception of a LIP F8 primitive sequence stays in bypass mode until the hub port receives a LIP primitive sequence other than a LIP F8 primitive sequence from the attached node port. In an alternative embodiment, upon receiving the LIP F8 primitive sequence the hub port, before entering bypass mode, passes a third current fill word to the next hub port in the loop.

The operation of a hub port in accordance with the preferred embodiment will be explained with reference to the components as illustrated in FIG. 3. Hub port 300 shown in FIG. 3 is used in a manner similar to a conventional hub port shown in FIG. 2A or 2B, such as hub ports 216 - 226, but has been modified as explained below.

An incoming internal hub link 302 enters hub port 300 and is connected to the output of a previous hub port (not shown). Incoming internal hub link 302 is connected to a hub port transmit circuit 304 and an input B of a switching device, such as a multiplexer 306. Hub port transmit circuit 304 receives data from incoming internal hub link 302 and passes that data to a node port 310 through a data channel 308 after converting the data to a form usable by node port 310. Node port 310 represents a connection to an operational device or a loop segment.

Node port 310, after performing any processing proper to its functionality and compliant with appropriate network protocols (*e.g.*, FC-AL protocols), transmits data back to hub port 300 through a data channel 312. Data channel 312 connects to a hub port receive circuit 314.

Hub port receive circuit 314 converts the data into a form usable in the hub. Hub port receive circuit 314 contains a loop initialization data detect circuit 316 and a hub port output control circuit 318. In an FC-AL implementation, loop initialization data detect circuit 316 is a LIP detect circuit. A hub port output control line 320 connects hub port output control circuit 318 to a control input of multiplexer 306. Hub port receive circuit 314 is connected to an input A of multiplexer 306 by a hub port output line 322. An input B of multiplexer 306 is connected to incoming internal hub link 302. A current fill word generator 324 is connected to an input C of multiplexer 306 through a current fill word line 326. The output of multiplexer 306 is connected to an outgoing internal hub link 328. Outgoing internal hub link 328 is connected to the input of the next hub port in the hub loop (not shown).

Under ordinary operations, when hub port 300 has an attached node port 310 which is operating properly and in compliance with network protocols such that loop failure initialization sequences are not being generated, hub port output control circuit 318 causes multiplexer 306 to select input A to be output to outgoing internal hub link 328. In this way, the output of node port 310 is passed to outgoing internal hub link 328. Alternatively, if no node port 310 is attached to hub port 300, hub port 300 is in a bypass mode. In bypass mode, hub port output control circuit 318 causes multiplexer 306 to select input B to be output on outgoing internal hub link 328. In this way, the data on incoming internal hub link 302 is passed directly to outgoing internal hub link 328 through multiplexer 306.

When loop initialization data detect circuit 316 detects that the data received by hub port receive circuit 314 from node port 310 is loop failure initialization data, loop initialization data detect circuit 316 sends a fill word flag to hub port output control circuit 318. In an FC-AL implementation, loop initialization data detect circuit 316 is a LIP detect circuit, as noted above. When LIP detect circuit 316 detects that the data received by hub port receive circuit 314 from node port 310 is a LIP F8 ordered set, LIP detect circuit 316 sends a fill word flag to hub port output control circuit 318. In response, hub port output control circuit 318 causes multiplexer 306 to select input C and pass a current fill word from current fill word generator 324 to outgoing internal hub link 328. If hub port receive circuit 314 receives a second consecutive identical LIP F8 ordered set, LIP detect circuit 316 keeps the fill word flag set. Hub port output control circuit 318 maintains the selection of input C of multiplexer

306, causing a second current fill word to be sent from current fill word generator 324 to outgoing internal hub link 328. If a second consecutive identical LIP F8 ordered set is not received, LIP detect circuit 316 clears the fill word flag. Hub port output control circuit 318 sets the selection of multiplexer 306 to input A, causing the data received by hub port

5   received circuit 314 from node port 310 to be output to outgoing internal hub link 328.

If a loop failure initialization sequence is received, loop initialization data detect circuit 316 sets a bypass flag. If the loop failure initialization sequence is not completed, loop initialization data detect circuit 316 clears the fill word flag and hub port output control circuit 318 selects input A of multiplexer 306. In response to the bypass flag, hub port output

10   control circuit 318 changes the input selection of multiplexer 306 to input B. The selection of input B of multiplexer 306 reflects the commencement of bypass mode for hub port 300. In an alternative embodiment, the selection of input B of multiplexer 306 is timed to occur after passing a third current fill word from current fill word generator 324 to outgoing internal hub link 328. In an FC-AL implementation, if a third consecutive identical LIP F8

15   ordered set is received, LIP detect circuit 316 sets the bypass flag. If a third consecutive identical LIP F8 ordered set is not received, the LIP F8 ordered set received flag is cleared by LIP detect circuit 316 and hub port output control circuit 318 selects input A of multiplexer 306.

Once hub port 300 has entered bypass mode due to the reception of a loop failure

20   initialization sequence from node port 310, hub port 300 maintains bypass mode until a recovery sequence is received by hub port receive circuit 314 from node port 310. Hub port receive circuit 314 also contains detection circuitry which detects the nature of data received by hub port receive circuit 314 (*i.e.*, whether the data is loop failure initialization data, a recovery sequence, *etc.*). When a recovery sequence (*e.g.*, three consecutive identical LIP

25   ordered sets of the same LIP type, which are not LIP F8 ordered sets) has been detected, the bypass flag is cleared. In response, hub port output control circuit 318 sets the input selection of multiplexer 306 to input A, connecting the output of node port 310 to outgoing internal hub link 328. In an FC-AL implementation, when a LIP primitive sequence other than a LIP F8 primitive sequence is detected, the bypass flag is cleared.

In an FC-AL implementation, LIP detect circuit 316 detects if any data received from node port 310 by hub port receive circuit 314 includes LIP ordered sets. If a LIP ordered set is detected, the LIP detect circuit then determines if the LIP ordered set is a LIP F8 ordered set or not. Once hub port 300 has entered bypass mode, the LIP detect circuit is used to

5 detect the reception of LIP ordered sets from the attached node port. When a LIP primitive sequence has been detected by the LIP detect circuit, and the received LIP primitive sequence is not a LIP F8 primitive sequence, the LIP detect circuit clears the bypass flag.

The automatic bypass of node port 310 upon detecting a loop failure initialization sequence from node port 310 conceals the occurrence of a data channel failure. The loop

10 operation continues without the complete collapse of loop operation as seen in FIG. 1A, 1B, 2A, and 2B. By replacing loop failure initialization data, such as the first two LIP F8 ordered sets received, by current fill words, unnecessary and possibly destructive loop failure initialization data is not introduced to the loop.

The preferred embodiment has been described along with several alternative

15 embodiments. However, variations which fall within the scope of the following claims are within the scope of the present invention. Accordingly, the present invention is not limited to the embodiment described above but only by the scope of the following claims.